

This listing of the claims with indicated amendments, deletions and cancellations and new claims will replace all prior version, and listings, of claims in the application.

LISTING OF THE CLAIMS

1. (cancelled)
2. (cancelled)
3. (cancelled)
4. (cancelled)
5. (cancelled)
6. (cancelled)
7. (cancelled)
8. (cancelled)
9. (cancelled)
10. (cancelled)
11. (cancelled)
12. (cancelled)
13. (cancelled)

PATENT

Amendment dated January 14, 2008
Reply to Office Action of September 12, 2007

Thought, Inc. Attorney Docket No.: 0036-023
Serial No. 10/046,894

14. (cancelled)

15. (cancelled)

16. (cancelled)

17. (cancelled)

18. (cancelled)

19. (cancelled)

20. (cancelled)

21. (cancelled)

22. (cancelled)

23. (cancelled)

24. (cancelled)

25. (cancelled)

26. (cancelled)

27. (cancelled)

28. (cancelled)

29. (cancelled)

30. (cancelled)

31. (cancelled)

32. (cancelled)
33. (cancelled)
34. (cancelled)
35. (cancelled)
36. (cancelled)
37. (cancelled)

38. (new) A local or distributed computer system for creating or maintaining transparent persistence to a permanent data source without the need for insertion of post compilation byte code into data objects of an object model or post compilation insertions of data objects into an object model or schema comprising:
 - (A) a software implemented method that utilizes at least one software component as a persistence manager,
 - (B) an associated programming interface, and
 - (C) a persistence layer for data source access to provide transparent persistence of a member selected from the group consisting of a complex data object, an object graph model and a portion of an object graph model, wherein:
 - (i) the complex data object, the object graph model or the portion of an object

graph model is persisted as a complex data object graph model,

(ii) the system comprises a user interface, a working memory storage area and at least one device for permanently storing information, and said system is loaded with at least one portion of a computer software program that comprises at least one user access interface, a programming object as a software component that is designed for creating or maintaining transparent persistence of a member selected from the group consisting of a data object, an object graph model and a portion of an object graph model as a complex data object graph model, and

(iii) the system further comprises a transparent persistence listing of a complex data object, an object graph model or a portion of an object graph model to be persisted by a transparent persistence method of the system, and items in the persistence listing are associated with a mapping layer set of tables or maps for objects and object attributes selected from the group consisting of relationships of objects to each other, relationships of objects to at least one data a source, and relationships of objects to an object graph,

software implemented methods for the persistence monitor of (A) comprise the logic to: (a) identify a complex object, object graph, or portion of an object graph residing on the system that is also associated with a persistence listing, (b)

detect any transactions that would require persistence of the complex object, object graph or portion of an object graph, and (c) delegate persistence to a data source functions to a persistence layer that handles persistence of such transactions to a permanent data source as a persisted complex object, object graph or portion of an object graph.

39. (new) A computer system according to claim 38, that does not require any modifications to an object model by modifying a post processed set of objects or a post processed object model to include additional persistent data objects or to include persistence byte code in a data object of the object model in order to provide persistence for all or a portion of the complex data object graph model.

40. (new) A computer system according to claim 39, that provides persistence for a complex data object model on a distributed network environment.

41. (new) A computer system for creating or maintaining transparent persistence to a permanent data source of a member selected from the group consisting of a complex data object, an object graph model and a portion of an object graph model when a user of the system is creating, maintaining, accessing or navigating complex

data objects as a complex data object graph model, comprising a software component as a loaded software module that is a loaded transaction monitor for accessing a persistence associated programming interface, wherein:

- (A) the loaded transaction monitor is associated with a persistence listing, an object application schema and data source schema mapping layer, and a runtime persistence layer for data source access to maintain persistence,
- (B) the loaded transaction monitor comprises logic or is associated with software implemented methods to inform a runtime persistence layer through a persistence associated programming interface that persistence as a complex data object graph model is needed for a complex data object, an object graph model, or a portion of an object graph, and
- (C) the loaded transaction monitor comprises logic or is associated with software implemented methods to determine that a runtime persistence layer notified through a persistence associated programming interface that persistence as a complex data object graph model is needed for a complex data object, an object graph model, or a portion of an object graph, persists the a complex data object graph about which it has been notified.

42. (new) A computer system according to claim 41, further comprising an entity java bean component, or similar functionality component, as a loaded software module.

43. (new) A computer system according to claim 41, that provides persistence for a complex data object graph model on a distributed network environment as an application without the need for server container managed persistence.

44. (new) A computer system as described in claim 38, comprising software logic for computer implemented methods, wherein such software logic is running in system memory or accessible by system memory and is an object programming language software module, or set of modules, and the computer implemented methods are for creating or maintaining transparent persistence, and wherein the transparent persistence is of a member selected from the group consisting of a data object, an object graph model and a portion of an object graph model when a user of the system is creating, maintaining, accessing or navigating complex data objects as a complex data object graph model.

45. (new) A computer system as described in claim 39, comprising software logic for computer implemented methods, wherein such software logic is running in

PATENT

Amendment dated January 14, 2008
Reply to Office Action of September 12, 2007

Thought, Inc. Attorney Docket No.: 0036-023
Serial No. 10/046,894

system memory or is accessible by system memory and is an object programming language software module, or set of modules, and the computer implemented methods are for creating or maintaining transparent persistence, and wherein the transparent persistence is of a member selected from the group consisting of a data object, an object graph model and a portion of an object graph model when a user of the system is creating, maintaining, accessing or navigating complex data objects as a complex data object graph model.

46. (new) A computer system as described in claim 40, comprising software logic for computer implemented methods, wherein such software logic is running in system memory or is accessible by system memory and is an object programming language software module, or set of modules, wherein the computer implemented methods are for creating comprising software logic for software implemented methods, and wherein the software running in system memory or is accessible by system memory is an object programming language software module, or set of modules, for creating or maintaining transparent persistence of a member selected from the group consisting of a data object, an object graph model and a portion of an object graph model when a user of the system is creating, maintaining, accessing